IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

INQUISIENT INC.,	)	
	)	
Plaintiff,	)	
	)	
v.	)	Civil Action No. 22-900-CJB
	)	
SERVICENOW, INC.,	)	
	)	
Defendant.	)	

MEMORANDUM OPINION AND ORDER

With regard to the parties' motions for claim construction, (D.I. 102; D.I. 104), the Court now addresses the construction of term 2: "type definition module."[1] The key dispute regarding this term is whether, during the prosecution of asserted United States Patent No. 7,979,468 (the "'468 patent"), Plaintiff disavowed the plain and ordinary meaning of the term, such that "type definition module" must instead mean "a non-standard object-oriented structure" that stores data types and metadata. (D.I. 100 at 31, 35; D.I. 112 (hereinafter, "Tr.") at 118-19) In order to invoke prosecution history disclaimer, Defendant must demonstrate that the patentee clearly and unambiguously disavowed claim scope. *See Aylus Networks, Inc. v. Apple Inc.*, 856 F.3d 1353, 1363 (Fed. Cir. 2017) ("[W]hen a prosecution argument is subject to more than one reasonable interpretation, it cannot rise to the level of a clear and unmistakable disclaimer.") (internal quotation marks and citation omitted). Here, for the following reasons, Defendant does not meet that bar.

[1] In doing so, the Court has considered the legal standards relating to claim construction. *See Vytacera Bio LLC v. CytomX Therapeutics, Inc.*, Civil Action No. 20-333-LPS-CJB, 2021 WL 4621866, at *2-3 (D. Del. Oct. 7, 2021).

1. First, by way of background regarding the parties' dispute, the Court notes some key relevant points on which the parties agree. It is undisputed that two different approaches to organizing and storing data within software are the relational model and the object-oriented model. (D.I. 101 at JA0000337, at ¶ 17; Tr. at 93-94) Nor is it disputed that relational database systems may still "utilize and incorporate object-oriented teachings and features." (D.I. 100 at 36 (citing D.I. 101 at JA0000342, at ¶¶ 41-43); *see also* Tr. at 107-08) And the parties also agree that: (a) according to the patentee, the patents "disclose a specific implementation of a conventional *relational* database comprised of database structures (or 'modules') that interrelate with each other[;]" but nevertheless (b) the claims include terminology (such as "class," "type" and "attribute") that "are borrowed from *object-oriented* programming." (D.I. 100 at 4, 44 (emphasis added); D.I. 101 at JA0000338-39, at ¶¶ 21-24, 29)

2. Defendant's disclaimer argument stems from Plaintiff's statements in the prosecution history regarding a prior art reference known as "Habela." The Examiner had initially rejected several proposed claims as being anticipated by Habela. (D.I. 101 at JA0000583) A subsequent interview between the Examiner and the patentee was summarized as follows:

> Inventor explained that the object[]-oriented teachings of the Habela reference do not read on the claimed invention *since the invention is not drawn to standard object-oriented structures*. The applicant intended to use *non-standard definitions for the normal terms such as the class and element that are used in the claims*. It was noted that the applicant should make clear where these definitions are found in the application so that the claims can be construed based on these non-standard definitions.

(*Id.* at JA0000593 (certain emphasis added, certain emphasis omitted)) In ultimately allowing the claims, the Examiner noted that the patentee distinguished those claims from Habela by stating that "certain portions of the claims drawn to interpretations of a class, type, and attribute, among other language in the claims, should 'not be defined as standard object-oriented structures' (Applicants 'Remarks' 1/21/2011, page 5)[.]" (*Id.* at JA0000367) And in the key portion of the patentee's remarks that the Examiner was citing to here, the patentee wrote that it had "presented arguments to the Examiner that the [] 'class attribute module' and the [] 'type definition module' *are not defined as standard object-oriented structures*." (*Id.* at JA0000586 (emphasis added))

3. The parties interpret this prosecution history statement of the patentee (and other related statements in the prosecution history) differently. In the end, however, the key issue is this: What was Plaintiff communicating when it told the Examiner—in order to gain allowance of its claims over Habela—that certain claim terms should not be defined "as standard object-oriented structures"? (Tr. at 103, 105, 112, 134) Plaintiff's position is that when it used those words (and particularly the word "structures"), it was distinguishing its invention by pointing out that Habela utilizes a "non-relational object oriented model" and "does not create any relational database schema." (D.I. 101 at JA0000584; *see also* D.I. 100 at 32, 41; Tr. at 97-99, 104-05) In other words, according to Plaintiff, it was simply telling the Examiner that while Habela was implemented as an object-oriented *database* (i.e., a "standard object-oriented structure[]"), its invention at issue was directed to and implemented as a relational *database*. (Tr. at 104-05) Moreover, Plaintiff asserts that it was *not* representing to the Examiner that "its patent claims must be wholly divorced from any *'teachings' or 'context'* that relates to the field of object-oriented programming or databases"—such as describing data used in terms of "class" or "element" or "attribute" or "type." (D.I. 100 at 41-42 (emphasis added); Tr. at 87, 107-08)[2]

[2] In the briefing, Defendant seemed to initially misunderstand Plaintiff's position in this regard. For example, Defendant stated in its answering brief that "InQuisient's apparent construction for this term is that it does not make use of any object-oriented teachings." (D.I.

Indeed, Plaintiff is arguing that its patents *do* make use of objected-oriented teachings and terminology in this fashion. (D.I. 100 at 44) Defendant, on the other hand, contends that when the patentee distinguished Habela on the ground that its invention did not implicate "standard object-oriented structures," the patentee was disclaiming the use in the patents-in-suit of any and all standard object-oriented *teachings* (which Defendant refers to, in part, as the use of "object-oriented metadata structures" such as "class, elements, attribute, and type")—even though the claims clearly include terminology borrowed from the object-oriented model. (*Id.* at 38-39, 46-47; Tr. at 124-26, 133-41)[3] And in light of this, Defendant argues that "type definition module" should be construed to mean "a non-standard object-oriented structure that stores data types and metadata by referencing data types defined by the 'element relation module,' 'class module,' 'attribute module,' 'class attribute module,' 'state machine module,' and/or 'tuple module.'" (D.I. 100 at 30, 38-39)

4. In the Court's view, Defendant's position too narrowly focuses on certain summary statements that the *Examiner* made, while sidestepping the content of the *patentee's* key assertions regarding why the claimed modules (like the type definition module) are not drawn to "standard object-oriented structures." (D.I. 101 at JA0000586-88; *see also* Tr. at 85, 107, 155-56, 160-63) For example, as noted above, in his Reasons for Allowance, the Examiner

100 at 36; *id.* at 39; *see also* Plaintiff's *Markman* Presentation Slides at Slide 27) But that was not Plaintiff's position—Plaintiff's view is that it never stated that its "claims eschewed '*any* object-oriented teachings.'" (D.I. 100 at 43 (emphasis added))

[3] While Defendant originally asserted that the alleged prosecution history disclaimer for this term rendered the term indefinite (because the person of ordinary skill in the art would not understand what a "non-standard" object-oriented structure was in the context of the asserted patents), Defendant agreed during the *Markman* hearing that any indefiniteness argument could be taken up at a later stage of the case. (Tr. at 142-43, 149) Ultimately, in light of the Court's conclusion that the substance of the patentee's prosecution history statements at issue here do not align with Defendant's view, Defendant's indefiniteness argument is moot.

summarized that "the applicants stated that certain portions of the claims drawn to interpretations of a class, type, and attribute, among other language in the claims, should 'not be defined as standard object-oriented structures'"—and in support, he cited to a specific page of the patentee's January 21, 2011 Remarks ("Remarks"). (D.I. 101 at JA0000367) Defendant latches onto this statement by the Examiner,[4] when it asserts that Plaintiff disclaimed all standard object-oriented structures, such as "class," "attribute" and "type." (D.I. 100 at 45, 47) But importantly, in the relevant, cited portion of the Remarks, the patentee seemed to be saying something different. There, the patentee appeared to make clear that its argument was that the "'type definition module'"—that is, the *module as a whole*—should not be defined as a "standard object-oriented structure[.]"[5] (D.I. 101 at JA0000586) In those Remarks, the patentee does not seem to be saying that the claims' utilization of words like "class," "type" and "attribute" alone should be drawn to or defined as non-standard object-oriented structures. (*Id.*) Indeed, the patentee went on to provide an "exemplary definition[]" for the "type definition module":

> [T]he claimed "type definition module" is a module that stores types of data and maintains the data language. . . . *[U]nlike a conventional relational database*, the [claimed elements metadata repository ("EMR")] extends the notion of available attributes types beyond the standard string, number, and date types. To do so, the EMR allows dynamically extendable entries *in a type definition table* 102. *Type definition table* 102 [which corresponds to the claimed "type definition module"] may store articulated types that may be, for example, an M4 layer while still maintaining the meta meta meta meta data language. By using the type of the person's name for an attribute, the user can declare more than what a *normal relational database* may store[.]

[4] Here the Examiner also noted that "for the purposes of the examination the appropriate language in the claims was construed according to the definitions found in the instant specification, as noted by the applicant." (D.I. 101 at JA0000367-68) It is notable that the language that the patentee construed was not "class," "attribute" or "type," but instead "type definition *module*" and "class attribute *module*." (*Id.* at JA0000586-87 (emphasis added))

[5] The patentee made the same point about the term "class attribute module," another disputed term that the Court will take up in turn. (D.I. 101 at JA0000586)

(*Id.* at JA0000587-88 (emphasis added); Tr. at 158-59) In that above paragraph, it seems like the patentee is distinguishing the invention from Habela based on the type of *relational database* that the invention makes use of (i.e., a database that uses tables). And then the patentee, pointing back to the above-referenced discussion, reiterated that "the claimed *modules* are not drawn to standard object-oriented *structures*." (D.I. 101 at JA0000588 (emphasis added)) This supports Plaintiff's position. That is, it bolsters Plaintiff's assertion that its Remarks at issue were directed to disclaiming the use of object-oriented *databases*, not the invention's use of any and all *teachings* bearing some relation to object-oriented databases. (D.I. 100 at 41-42; *see also* Plaintiff's *Markman* Presentation Slides at Slide 31; Tr. at 158-59)[6] At a minimum, it certainly does not suggest that the patentee made a "clear and unmistakable" disclaimer of the type argued by Defendant. (D.I. 100 at 44)

5. Since Defendant's flawed prosecution history disclaimer argument drives its proposed construction, (*see, e.g.*, *id.* at 35) the Court will not adopt that proposal. Nor will it adopt Plaintiff's proposal ("a module that can define and store data types"), since it is redundant of the claim language itself. (*See* '468 patent, col. 16:34-35 ("a type definition module configured to define and store one or more types of the class"); Tr. at 83, 117)

[6] The patentee also provided an exemplary definition for "class attribute module." (D.I. 101 at JA00000587) It explained that the class attribute table described in the specification "allows for one attribute to be referenced by many classes [such that] there is attribute inheritance among the classes" and it notes that "[a]ttribute inheritance among the classes is a powerful notion not found in traditional relational database management systems[.]" (*Id.* (emphasis omitted)) Defendant's expert observes that the patents describe the class attribute table using terms found in object-oriented teachings. (*Id.* at JA0000339, at ¶ 29) Thus, the patentee seemed to be acknowledging here that the patents utilized certain teachings from the object-oriented model (and was not disclaiming any and all teachings relating to object-oriented programming). (*See* Tr. at 107-08; Plaintiff's *Markman* Presentation Slides at Slide 36)

6. Having herein resolved the parties' key dispute with respect to this term, and with there otherwise being no additional clear ripe disputes, the Court concludes that the term should simply be afforded its plain and ordinary meaning. *See, e.g.*, *Finjan, Inc. v. Secure Computing Corp.*, 626 F.3d 1197, 1207 (Fed. Cir. 2010) (concluding that the district court's rejection of the defendant's construction and the court's adoption of "plain and ordinary meaning" as to a claim term properly resolved the dispute between the parties).

Dated: July 31, 2024

Christopher J. Burke
Christopher J. Burke
UNITED STATES MAGISTRATE JUDGE